

Closed Loop Flow Control Workshop 2005
Jackson Hole, Wyoming, USA, July 19, 2005

A Multiscale and Multifidelity Framework for Simulating Flow Control Systems

Scott Collis

Computational Mathematics & Algorithms

Sandia National Laboratories

Albuquerque, NM 87185-1110

sscoll@sandia.gov

Joint work with:

Guoquan Chen & Srinivas Ramakrishnan

Mechanical Engineering and Materials Science

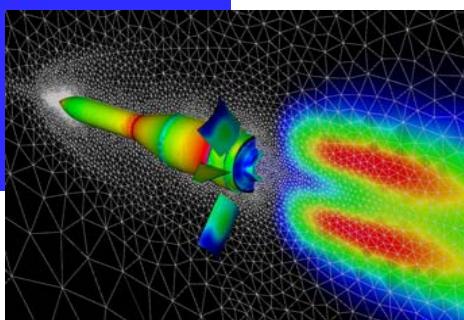
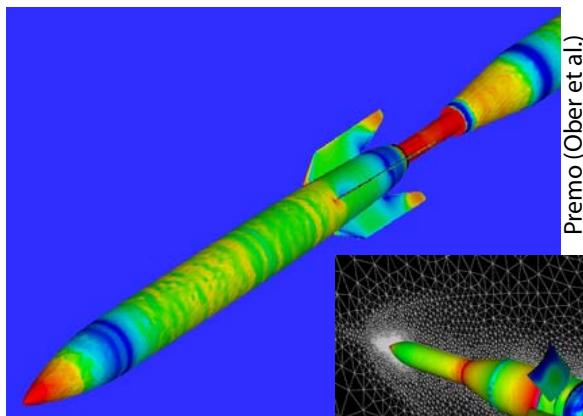
Rice University, Houston, TX 77005

gqchen@rice.edu

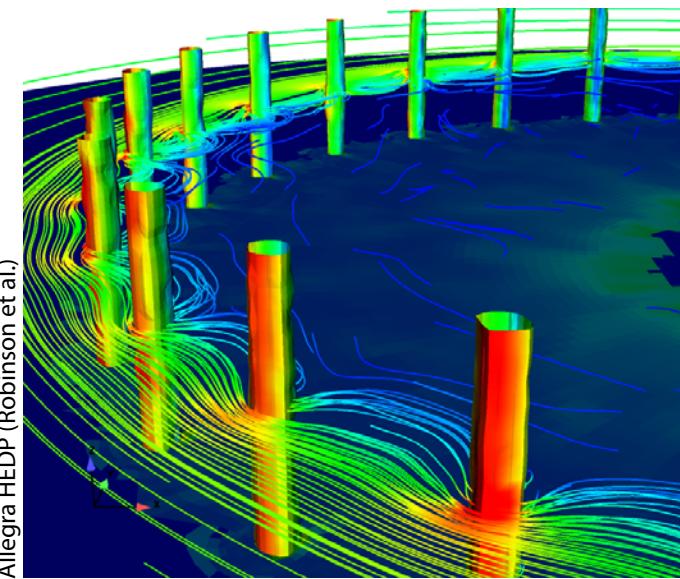
Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under contract DE-AC04-94AL85000.



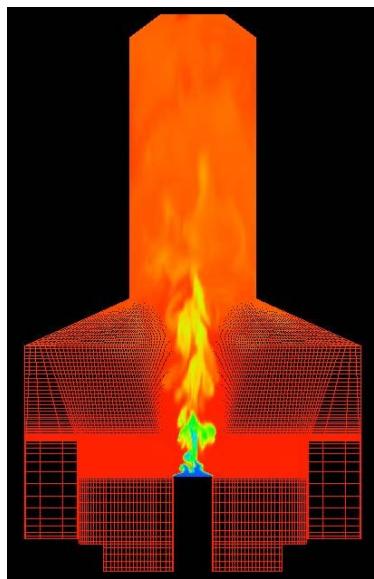
Sandia and Fluid Flows



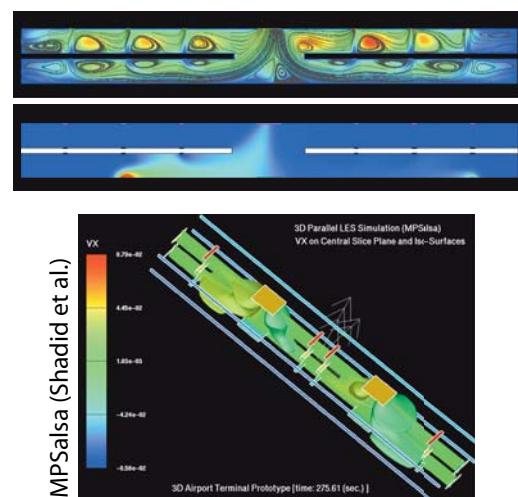
Aerodynamics



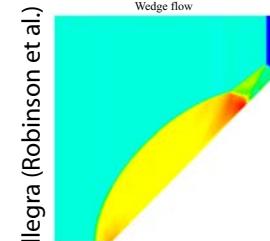
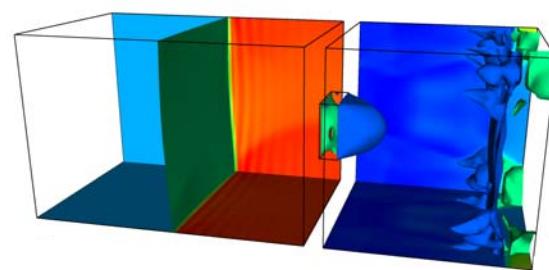
MHD



Combustion

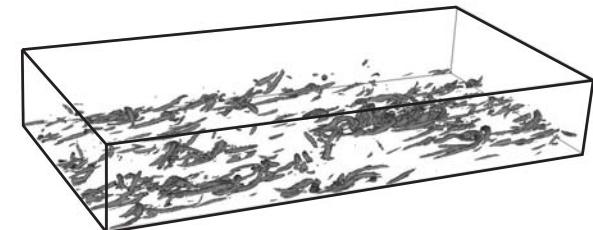
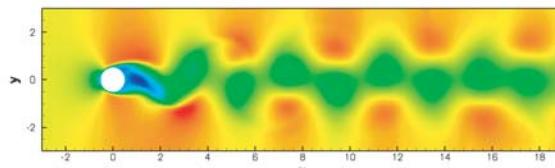


Homeland Security



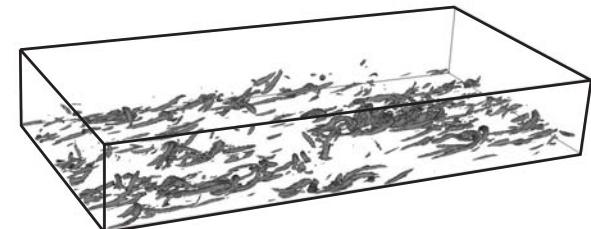
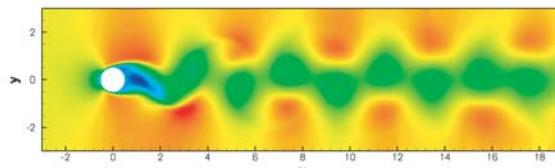
Shock Hydrodynamics

Optimization of Unsteady *Compressible* Flows



- Application of optimization to transient, *compressible* flows is largely untapped...
- Transient optimization and control problems are increasingly important:
 - Steady-state solutions do not capture critical physics:
aeroacoustics, combustion instabilities, shock/BL interaction, wakes, ...
 - Next generation systems will likely use *active* design/control techniques.
- Algorithmic Challenges:
 - Complex geometries
 - Unsteady flow physics
 - Localized, broadband physics
 - Gradient evaluation
 - Storage of time-history
 - Complex problem setup
 - Large-scale space-time problems

Optimization of Unsteady *Compressible* Flows



- Application of optimization to transient, *compressible* flows is largely untapped...
- Transient optimization and control problems are increasingly important:
 - Steady-state solutions do not capture critical physics:
aeroacoustics, combustion instabilities, shock/BL interaction, wakes, ...
 - Next generation systems will likely use *active* design/control techniques.
- Algorithmic Challenges:
 - Complex geometries
 - Unsteady flow physics
 - Localized, broadband physics
 - Gradient evaluation
 - Storage of time-history
 - Complex problem setup
 - Large-scale space-time problems
- DG + Optimization = SAGE
 - Unstructured meshes.
 - High accuracy, low-dissipation.
 - Multiscale / zonal models.
 - Adjoint methods.
 - Efficient I/O, checkpointing, TDD
 - Object-oriented software design.
 - Parallel algorithms.



Discontinuous Galerkin

Key Points:

- A hybrid between finite element and finite volume methods.
- Solutions continuous in elements, discontinuous across element interfaces.
- Elements are coupled via *numerical fluxes* on element interfaces.

Advantages:

- Spectral accuracy on arbitrary meshes,
- Local hp -refinement,
- Diagonal mass matrix,
- Weak boundary conditions,
- Local conservation,
- Low communication overhead.

Disadvantages:

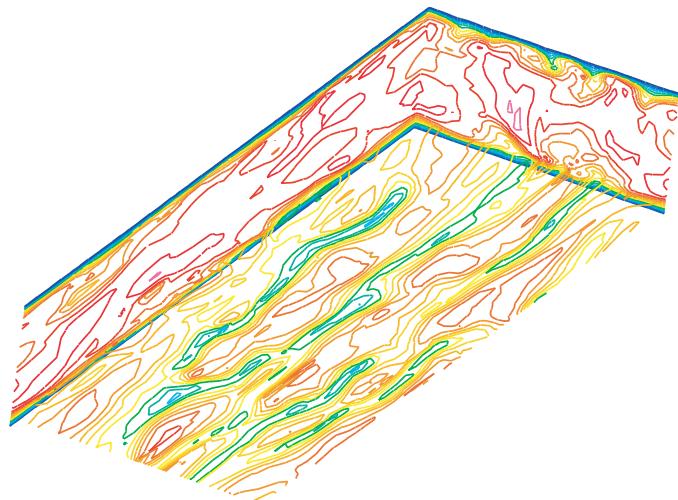
- More unknowns for same accuracy,
- Potentially high FLOP count,
- Aliasing at high-orders,
- Requires high-fidelity geometry,
- Limiters required for shocks ($p > 0$),
- *Must* exploit inherent flexibility.

Bottom-line:

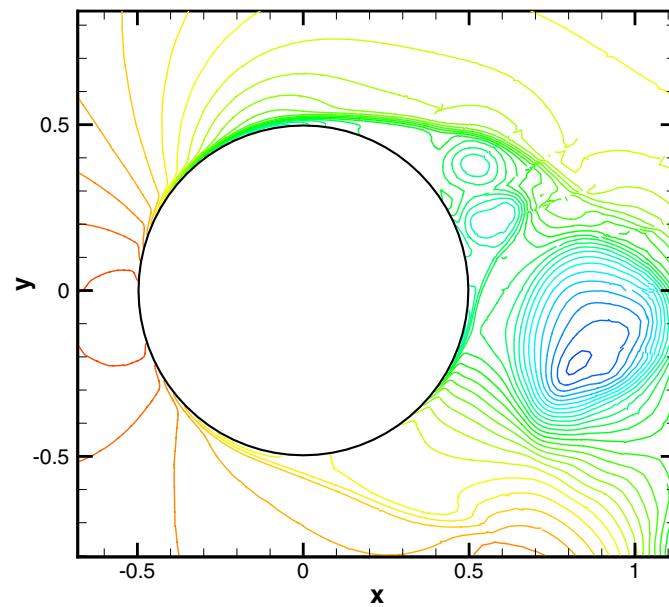
High accuracy, high flexibility make DG particularly promising for transient, multi(**scale, model, fidelity**) optimization problems, *but there is no free-lunch...*

Status of SAGE Flow Solver

- Arbitrarily high-order discontinuous Galerkin spatial discretization,
- Explicit and implicit time-advancement (checkpointing on the way)
- Designed to support transient optimization problems,
- Supports **multifidelity** and **multiscale** models,
- ℓ VMS approach for element-by-element subgrid-scale modeling,
- Already validated for a variety of **laminar** and **turbulent** flows ...



with Srinivas Ramakrishnan (Rice)



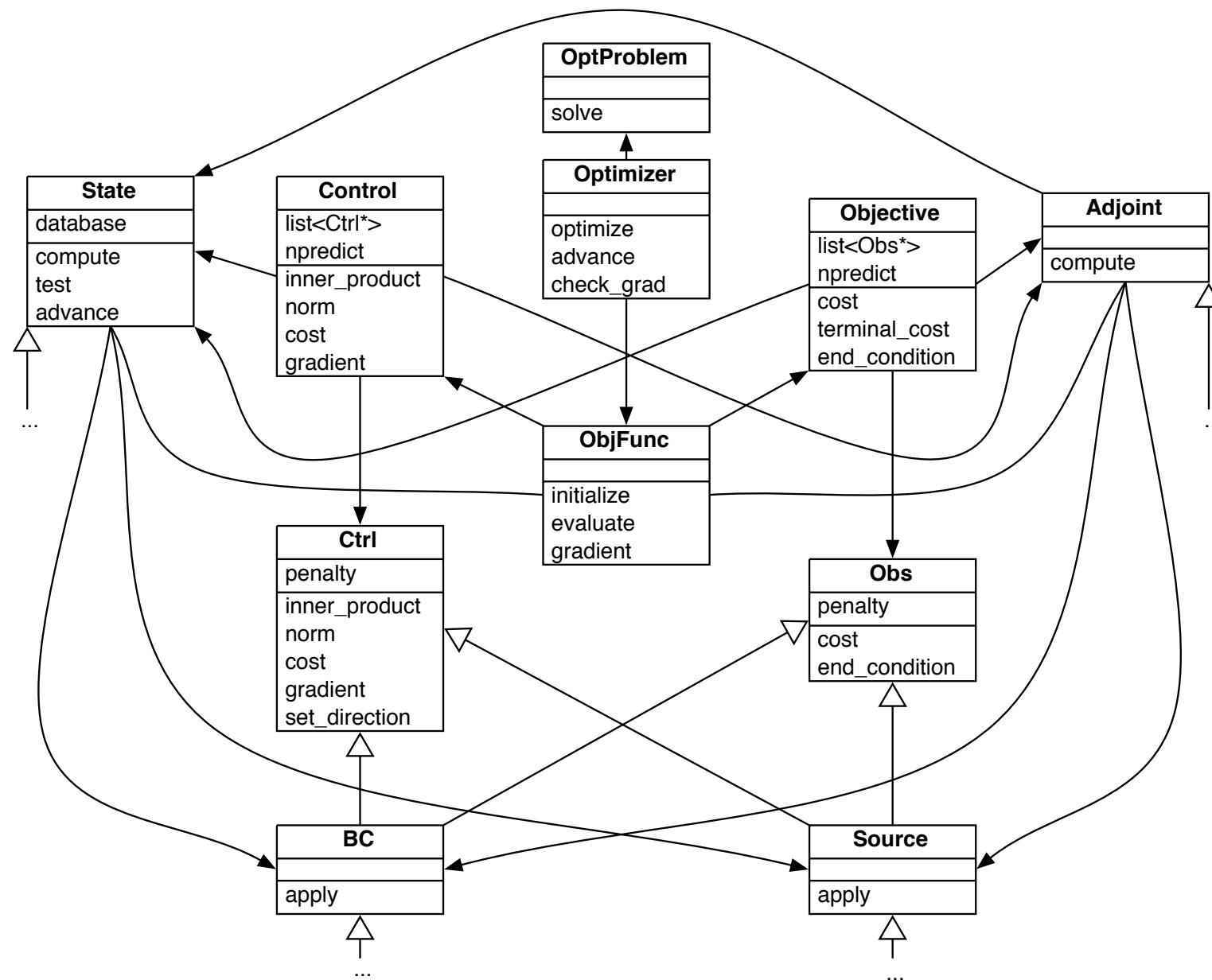
with Matt Barone (Sandia)



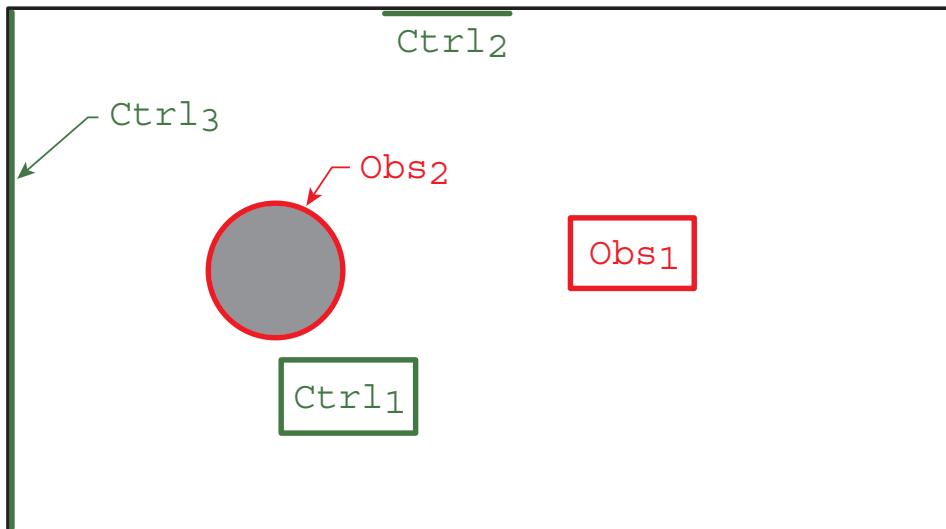
DG + Optimization: SAGE

- Continuous (and/or discrete) adjoint formulation in space:
 - Adjoint PDEs are discretized using DG, similar to State equations.
 - Allows for accurate, stable, discretizations of both state and adjoint.
 - Enables different resolutions to be used for state and adjoint.
 - Obviates difficulties with non differentiable numerical fluxes and limiters common in DG (and FV) discretizations.
 - Provides insight into the physics of sensitivity systems and boundary conditions.
 - **With particular choice of adjoint flux, can also be discrete adjoint at same resolution!**
- Discrete adjoint in time: Runge-Kutta, Backward Euler, Midpoint
- Supports: Advection-Diffusion, Burgers, Wave, Euler, and Navier-Stokes
- Can utilize Sandia's DAKOTA and MOOCHO optimization tools...
- Allows for time-domain decomposition techniques
(Bartlet, Collis, Heinkenschloss, van Bloemen Waanders, 2004)...
- Generic solver/optimization interface mimics mathematical formulation and optimization problem setup ...

SAGE Framework Design



Modeling Systems of Sensors and Actuators



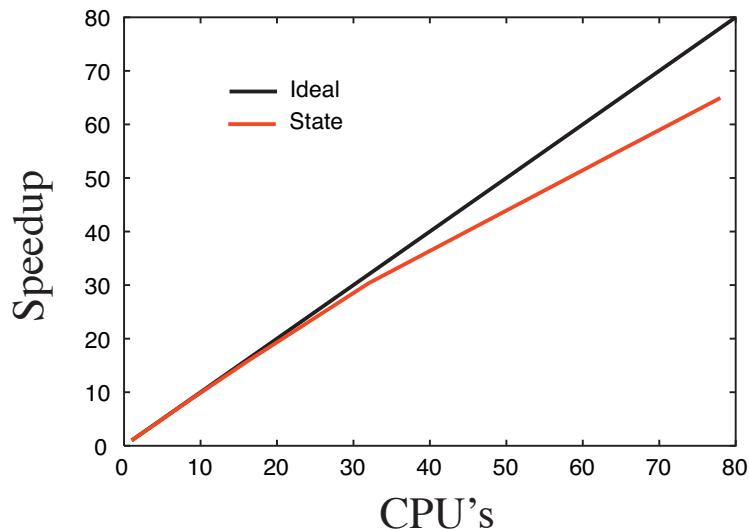
- Supports multiple **Obs** and **Ctrl** objects

$$J(\mathbf{y}, \mathbf{u}) = \sum_{n=1}^{N_{\text{obs}}} J_{\text{obs}_n}(\mathbf{y}) + \sum_{m=1}^{N_{\text{ctrl}}} J_{\text{ctrl}_m}(\mathbf{u})$$

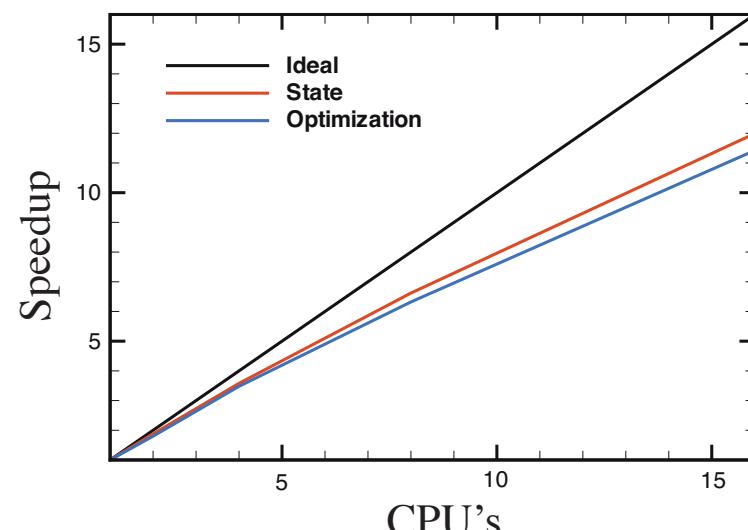
- Each **Obs** and **Ctrl** are *self-contained* and interact with the **State** and **Adjoint** as in continuous adjoint formulation.
- Design makes adding new **Obs** and **Ctrl** straightforward.

SAGE Implementation

- C++ with STL and all kernel computations using ATLAS/BLAS.
- Supports: Advection-Diffusion, Burgers, Wave, Euler, Navier-Stokes
- Designed from the ground up for parallel execution using MPI-2 and MPI-IO.
- Excellent scaling for flow solver, even on small 2-d problems ($\approx 10,000$ elements).
- **Scaling of optimization problems same as flow solver, so far...**
(Note that optimization problem is small — **only 576 elements!**)



Parallel speedup of DGM Solver.



Parallel speedup of DGM-Opt.

- Runs on: Linux, Mac OS-X, Cygwin, Alpha, SGI, most Sandia/DOE platforms.



Sandia Enabling Technologies

SAGE leverages Sandia's Object Oriented Toolkits

- **Trilinos (Heroux, et al.)**

- Distributed, parallel sparse vectors and matrices
- Time discretization
- Spatial discretization (coming soon)
- Nonlinear solvers (Newton, quasi-Newton, Picard, ...)
- Linear solvers (Parallel direct, Krylov, ...)
- Eigensolvers (parallel, iterative)



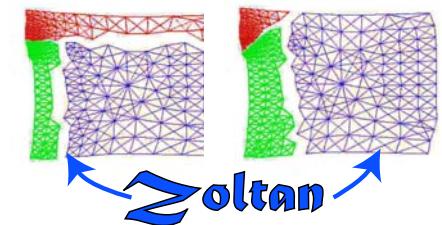
- **Dakota (Eldred, et al.)**

- Gradient based optimization: NCG, SQP, BFGS, ...
- Genetic algorithms, direct search
- Surrogate-based optimization
- Multifidelity optimization
- Uncertainty quantification



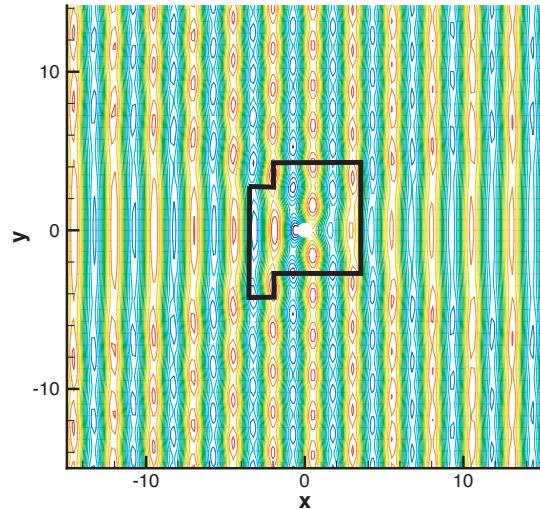
- **Zoltan (Devine, et al.)**

- Graph and hyper-graph parallel partitioners
- Dynamic load-balancing
- Data migration and redistribution

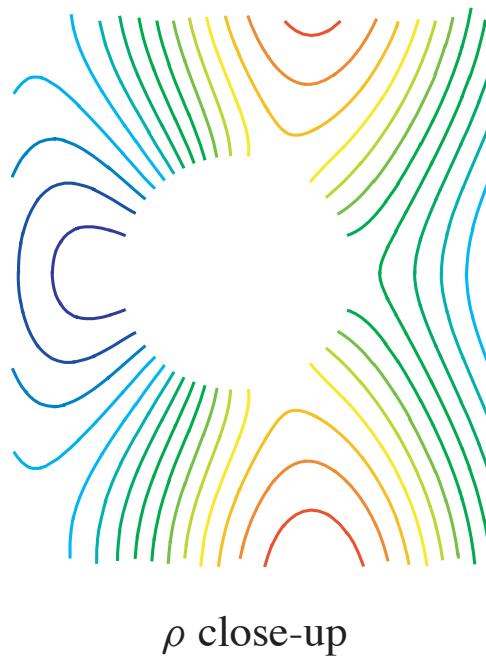


Acoustic Scattering from a Circular Cylinder

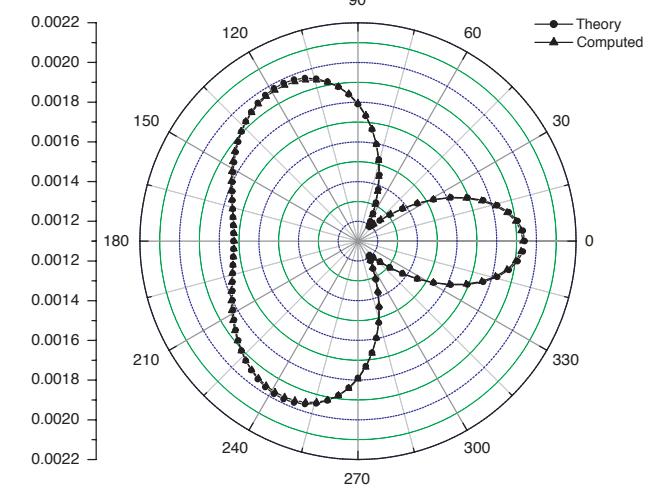
- Conditions: zero mean flow, incident planar acoustic wave, $\lambda/d = 2.5$.
- Discretization: 6832 quadrilaterals with $p = 6$, sponge layer on farfield.
- Models: Euler equations near the cylinder, wave equation in the farfield



Incident + Scattered Pressure



ρ close-up

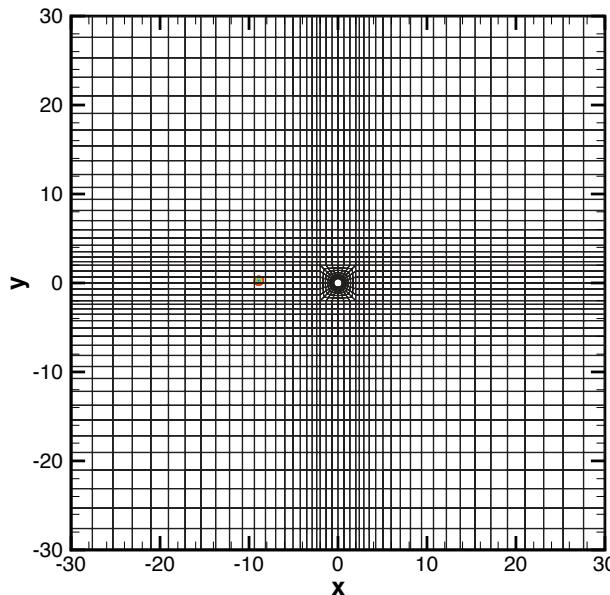
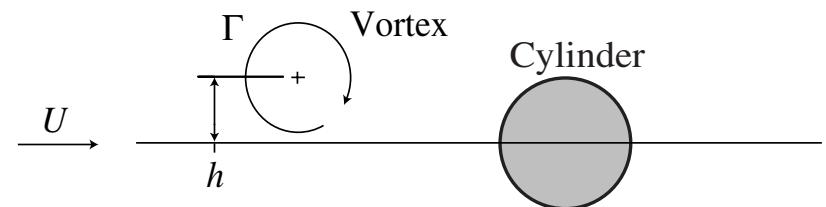


Acoustic Amplitude

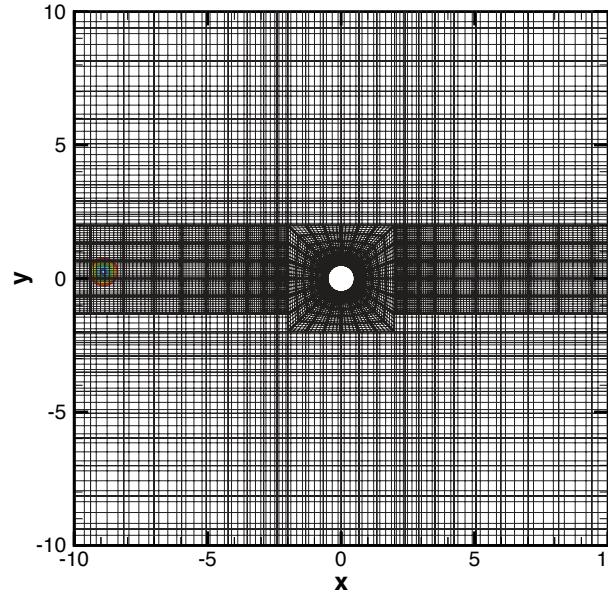
- Excellent agreement with theory
- Note clean density near solid boundaries — this is *very* hard to achieve with typical high-order aeroacoustics codes...

Cylinder-Vortex Interaction

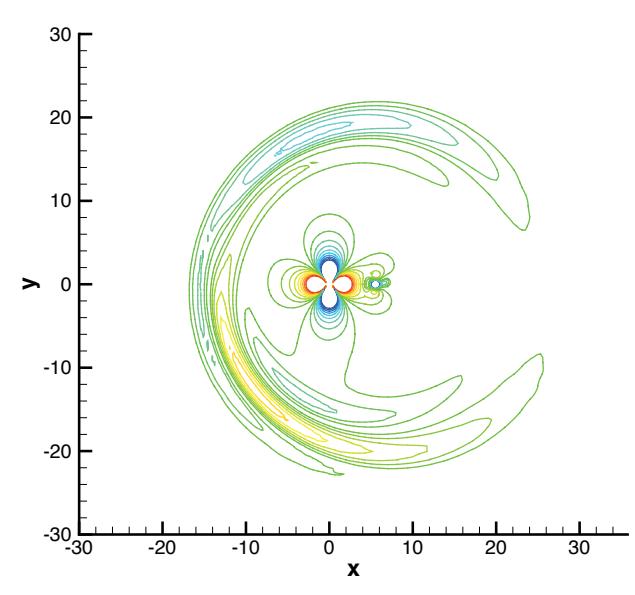
- Conditions: inviscid, $M_\infty = 0.3$ mean flow.
- Vortex: Location $(x_0, y_0) = (-9.0, 0.25)$, core radius $R_c = 0.4$, maximum velocity $v_{\theta \max} = 0.5$.
- Discretization: 2224 quadrilaterals, hybrid polynomial order $p = 5$ in the vortex path, $p = 3$ elsewhere.



Mesh and vortex.



Close up of mesh and vortex.

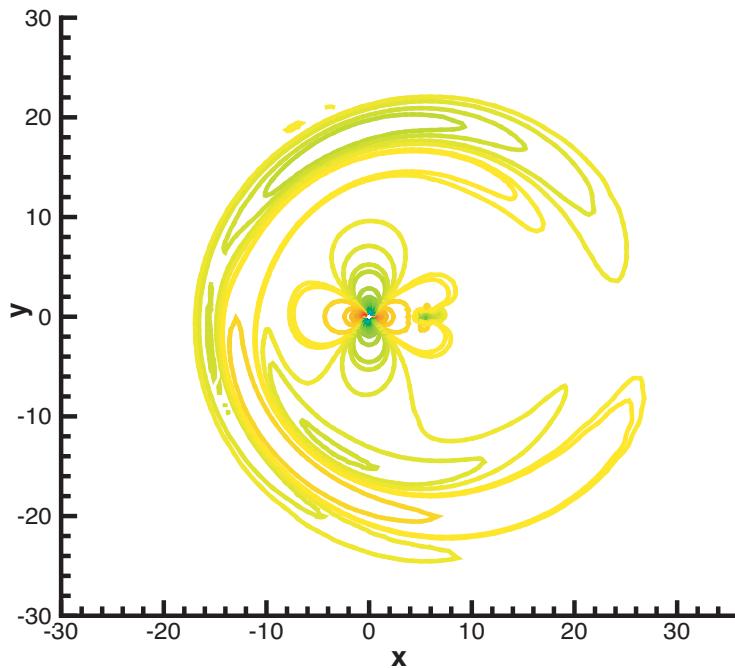
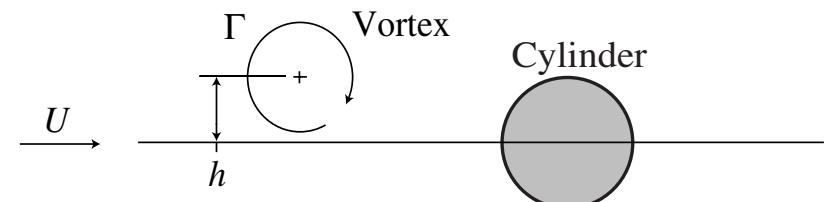


Pressure

Cylinder-Vortex Interaction

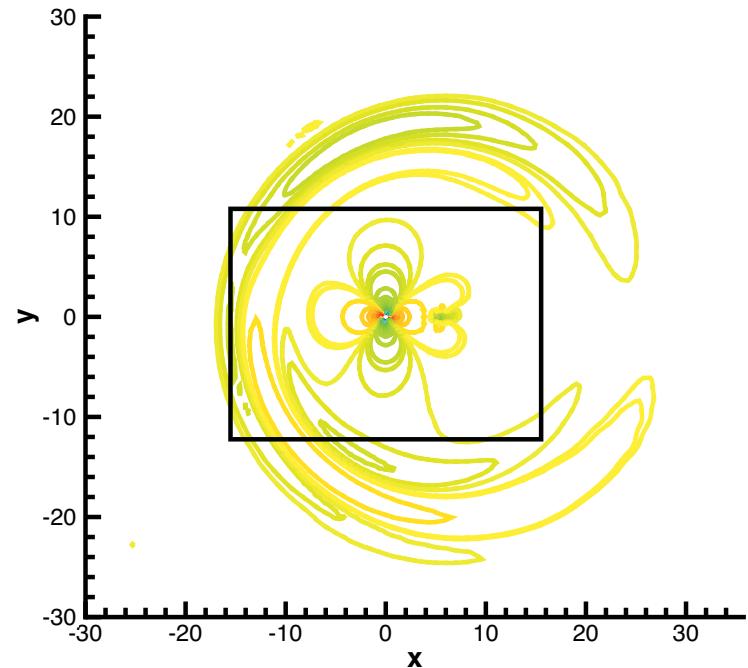
(Euler + Wave Equation)

- Conditions: inviscid, $M_\infty = 0.3$ mean flow.
- Vortex: location $(x_0, y_0) = (-9.0, 0.25)$,
core radius $R_c = 0.4$, maximum velocity
 $v_{\theta \max} = 0.5$.



Euler

Pressure

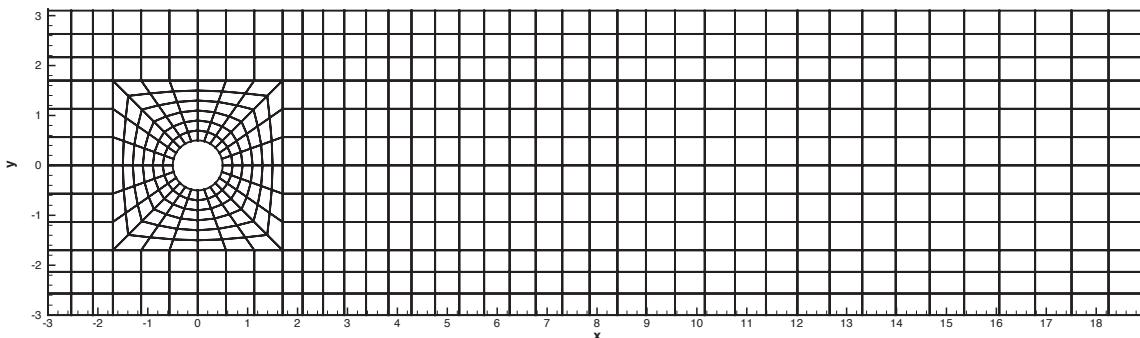
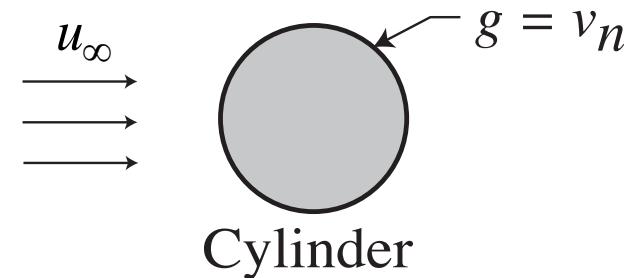


Euler + Wave Eqn.

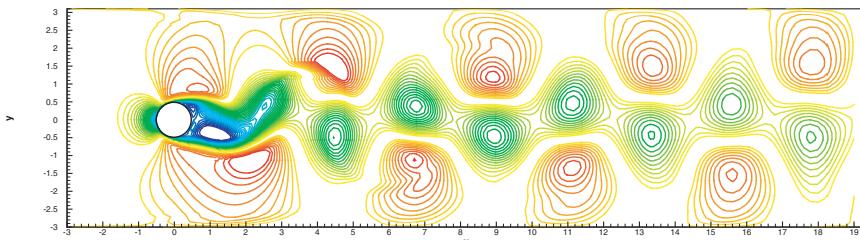
Pressure

Cylinder Wake Control

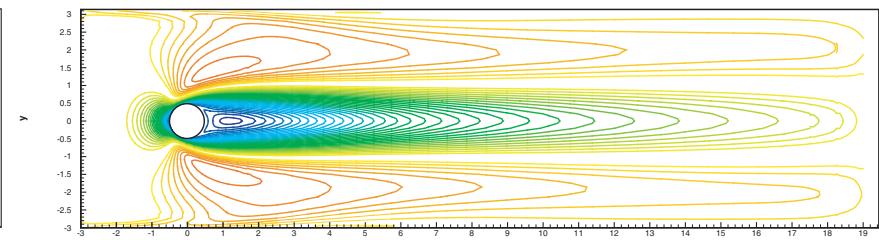
- Conditions: $Re = 100, M_\infty = 0.5$.
- DG using $N_e = 576$ quadrilaterals with $p = 4$; RK4 in time: $N_t = 2,000, \Delta t = 0.0015$.
- Consider both *unsteady* and *steady* suction/blowing.
- Full State tracking objective.



Element mesh



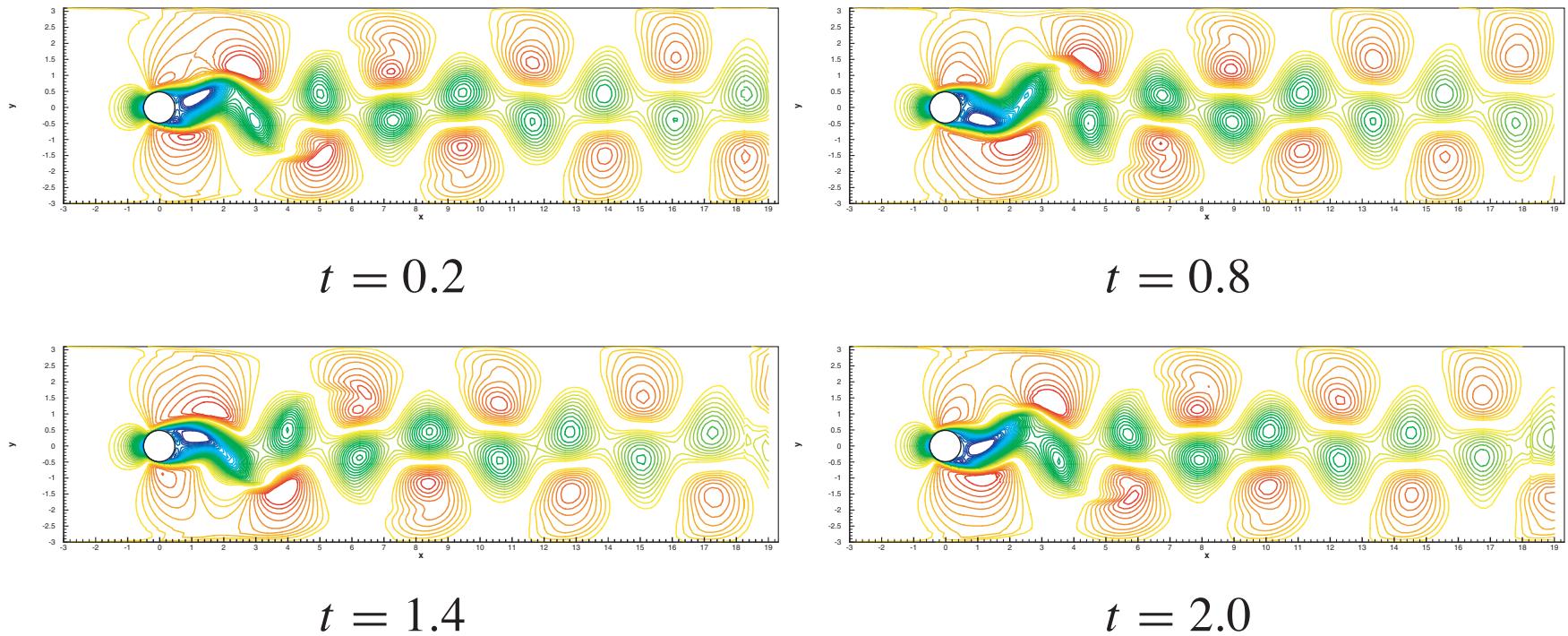
Uncontrolled ρu at $Re = 100$.



Uncontrolled ρu at $Re = 20$

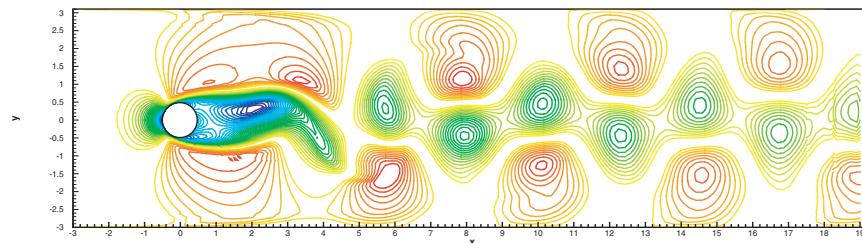
Cylinder Sheding: Streamwise Momentum ρu

No Control

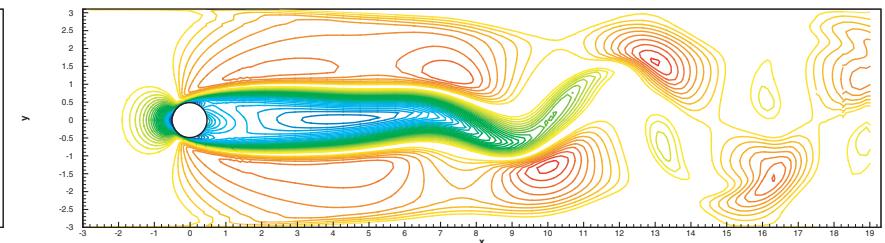


Cylinder Sheding: Streamwise Momentum ρu

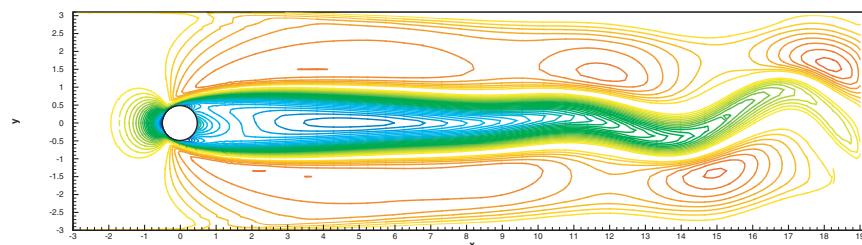
Optimal Steady Control



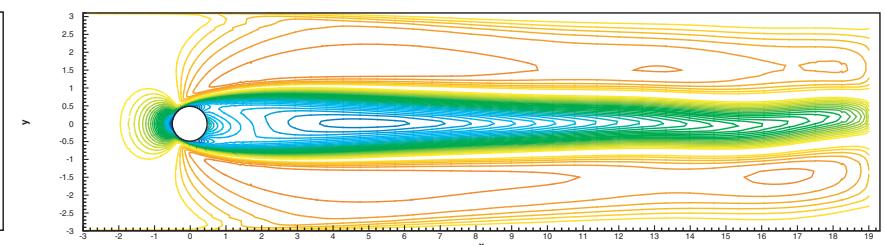
$t = 2.0$



$t = 14.0$

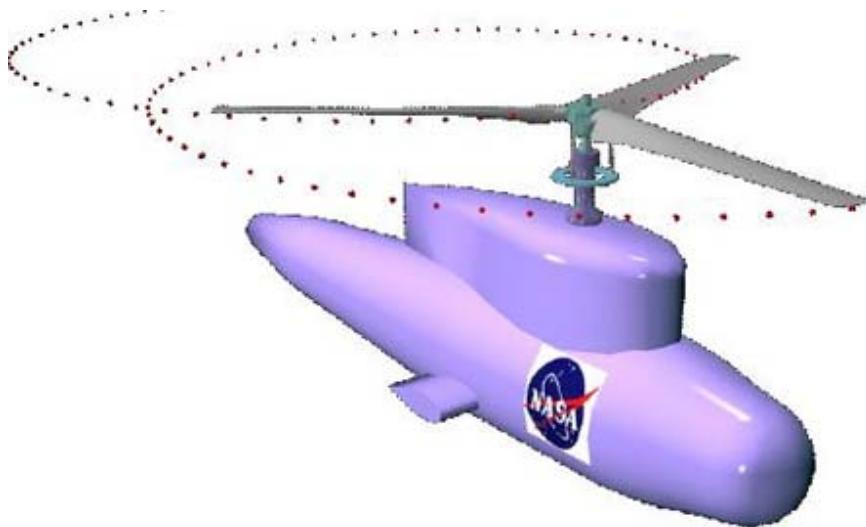


$t = 30.0$

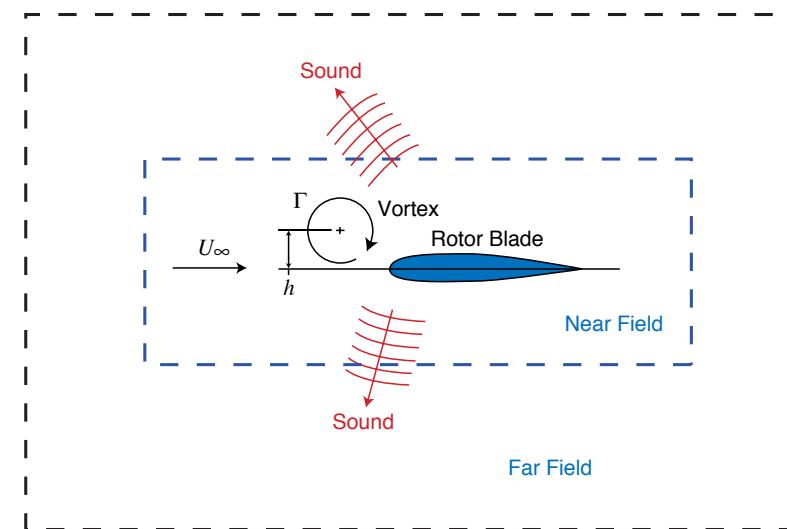


$t = 40.0$

Blade-Vortex Interaction



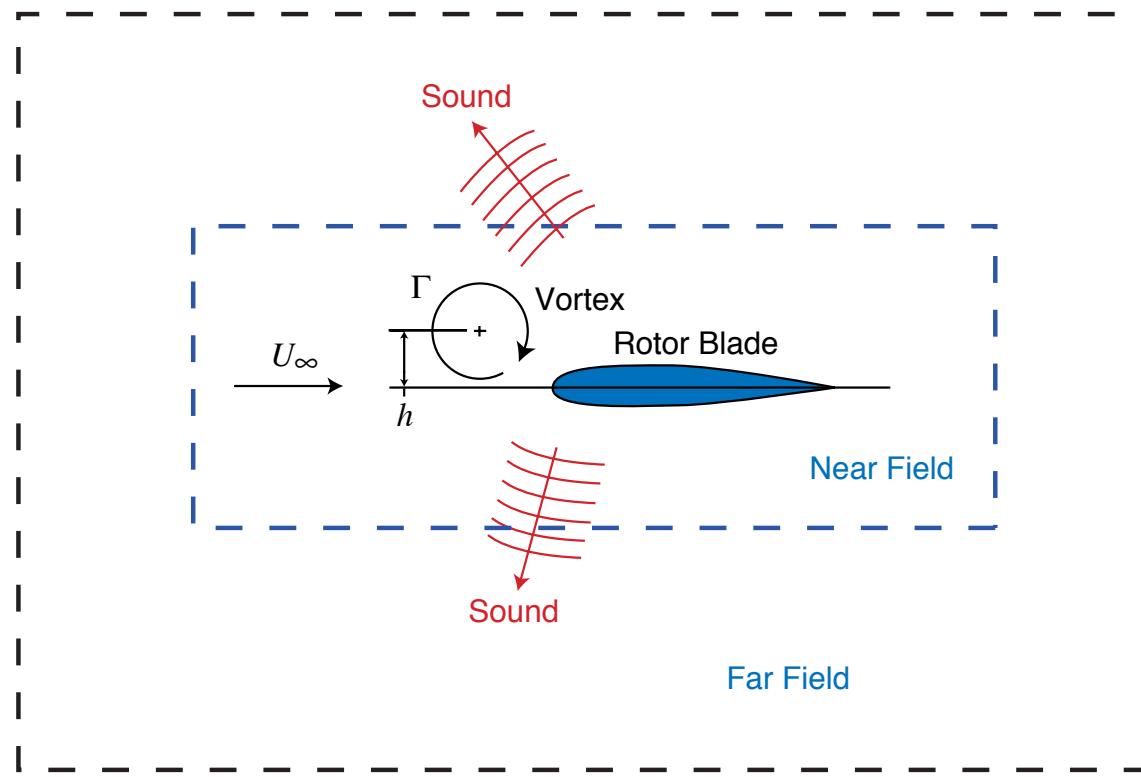
Parallel BVI (NASA Ames)



Multimodel approach

- BVI occurs in low speed descending flight conditions. results in **high amplitude** impulsive noise that radiates towards the ground.
- Can optimal control lead to new techniques that alleviate BVI noise?
- Optimal control of unsteady compressible flows is *very* challenging...

Optimal Control of Multi-Model Systems



$$\text{Min } J(\mathbf{y}, g)$$

such that

$$\mathcal{N}(\mathbf{U}(\mathbf{Y}), g) = 0 \text{ in } \Omega_{\text{near}}$$

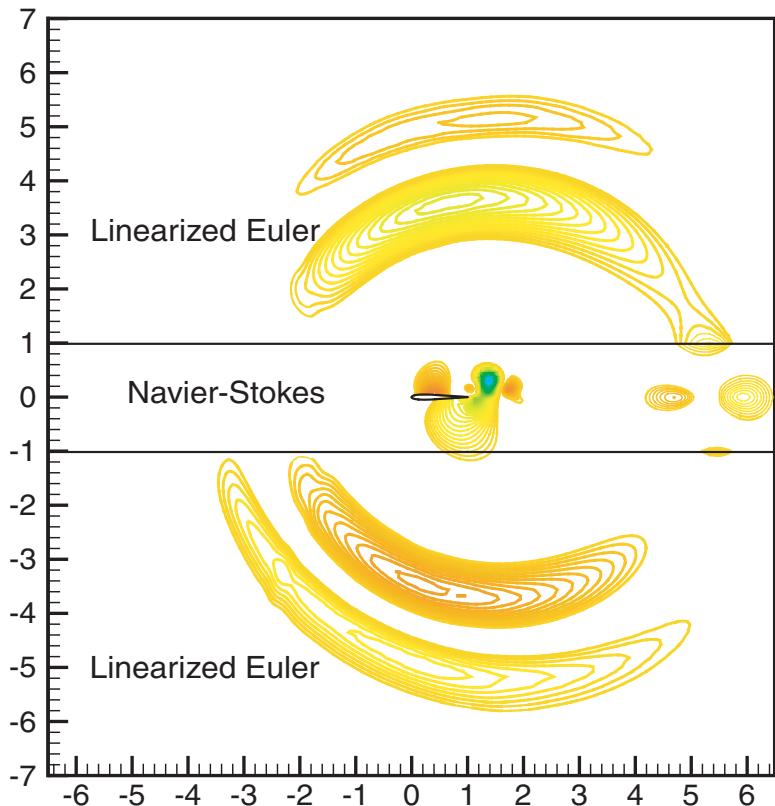
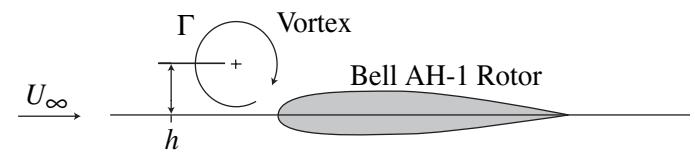
$$\mathcal{F}(\bar{\mathbf{Y}}, \mathbf{y}) = 0 \text{ in } \Omega_{\text{far}}$$

- $\mathcal{N}(\mathbf{U}(\mathbf{Y}), g) = 0$ near-field equations
- $\mathcal{F}(\bar{\mathbf{Y}}, \mathbf{y}) = 0$ far-field equations
- \mathbf{U} near-field flow variables
- \mathbf{y} far-field fluctuation variables
- g on-blade control

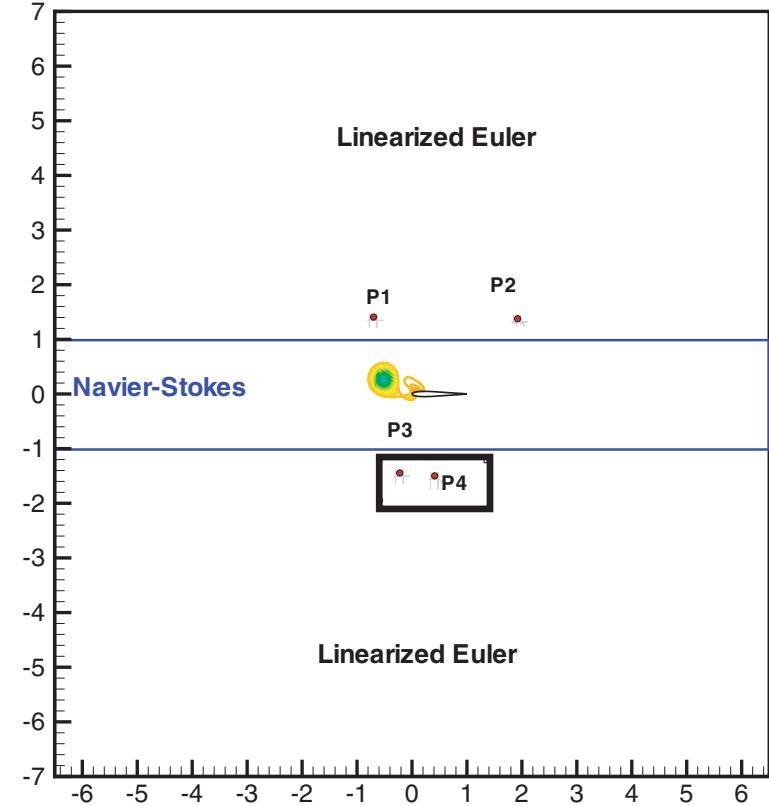
Multimodel Simulation for Optimal Control of BVI

(Navier–Stokes & Linearized Euler Equation)

- Conditions: viscous, $M_\infty = 0.3$ mean flow.
- Vortex: location $(x_0, y_0) = (-6.0, 0.25)$,
core radius $R_c = 0.15$, maximum velocity
 $v_{\theta \max} = 0.5$.



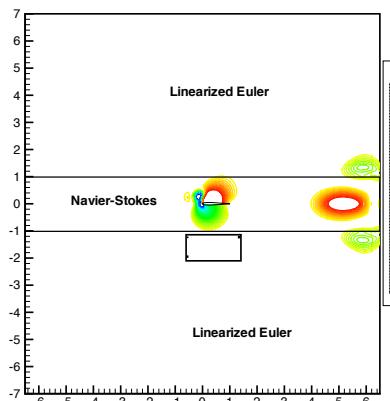
NS & LE: Scattered Pressure.



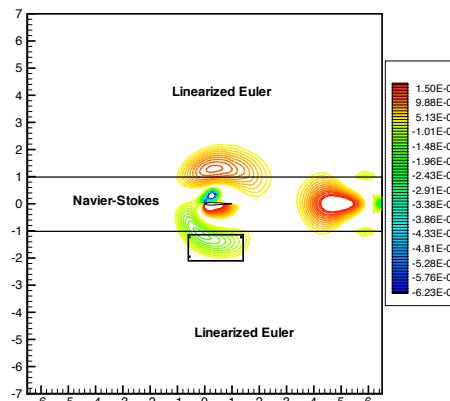
Problem setup for BVI

Scattered Pressure Contours

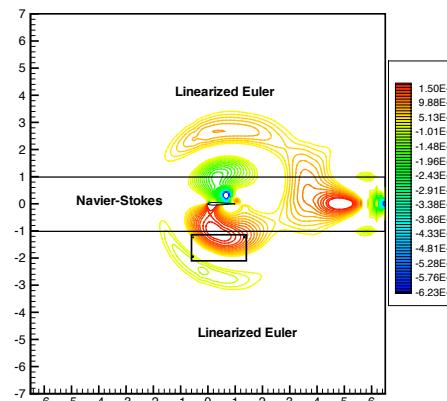
No Control



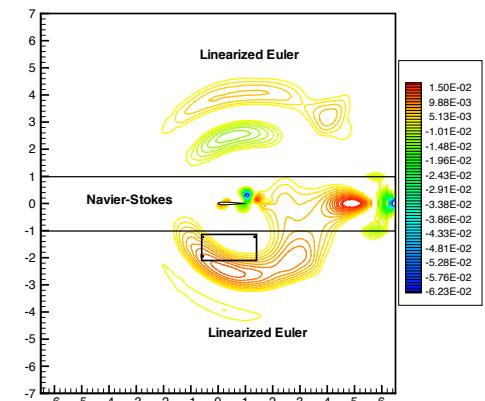
$t = 1.76$



$t = 2.16$

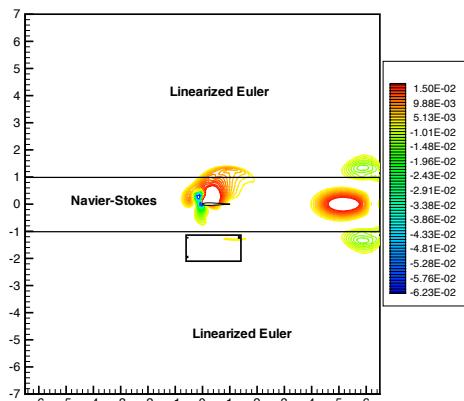


$t = 2.56$

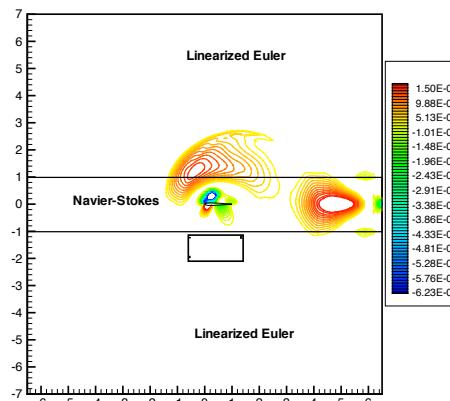


$t = 2.96$

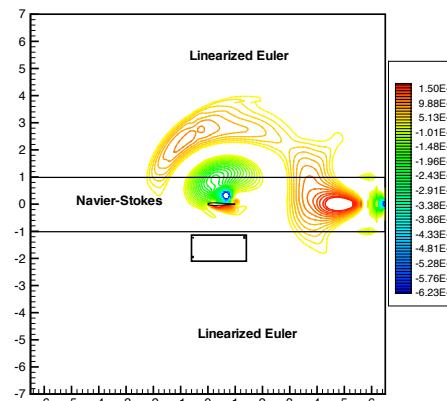
Optimal Control



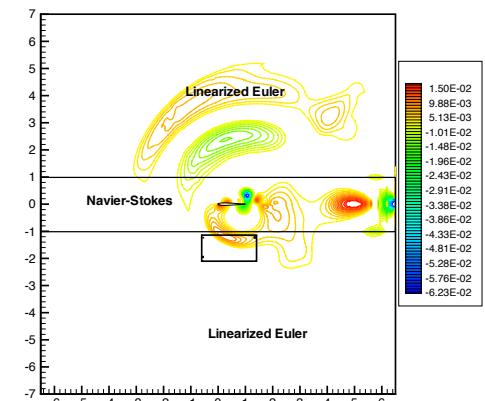
$t = 1.76$



$t = 2.16$



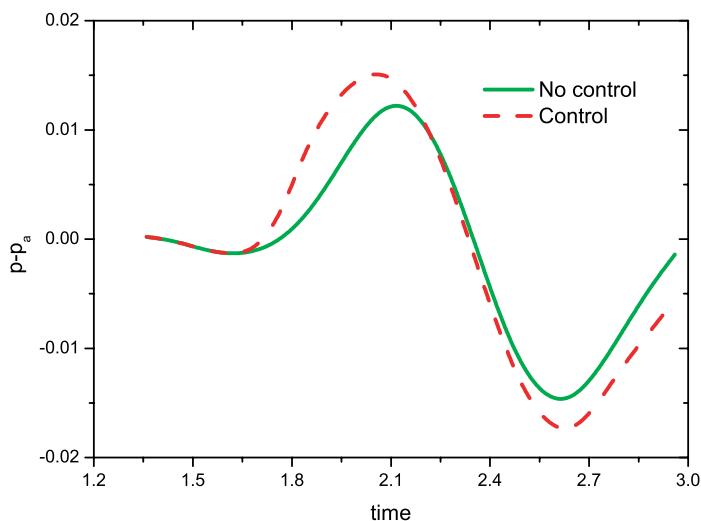
$t = 2.56$



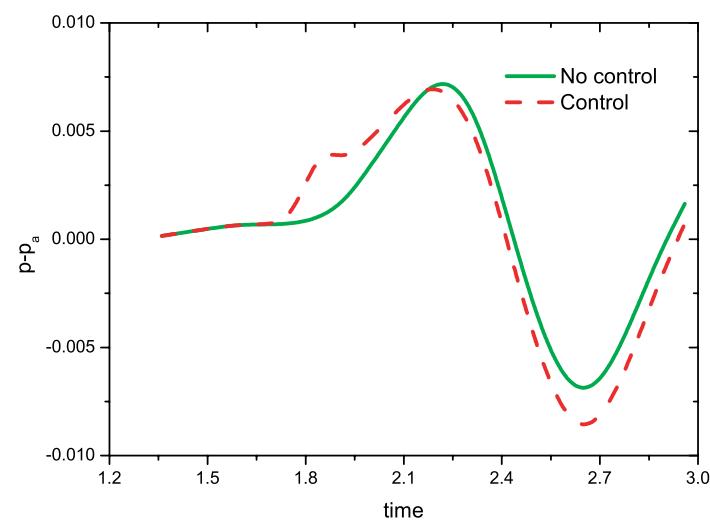
$t = 2.96$



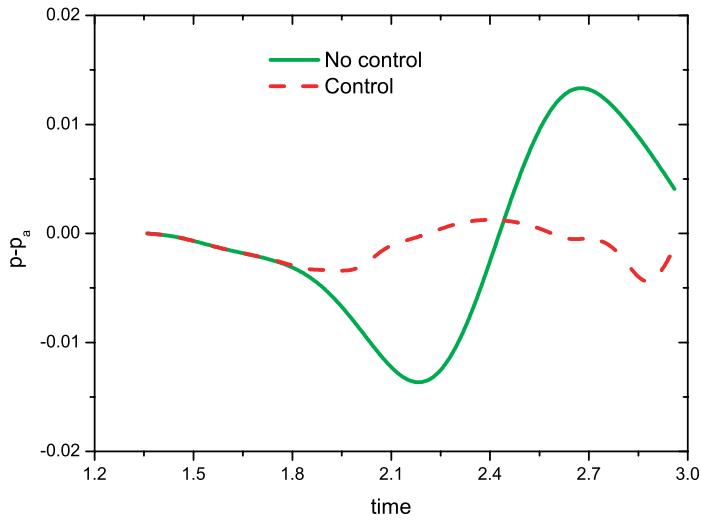
Scattered Pressure Time Histories



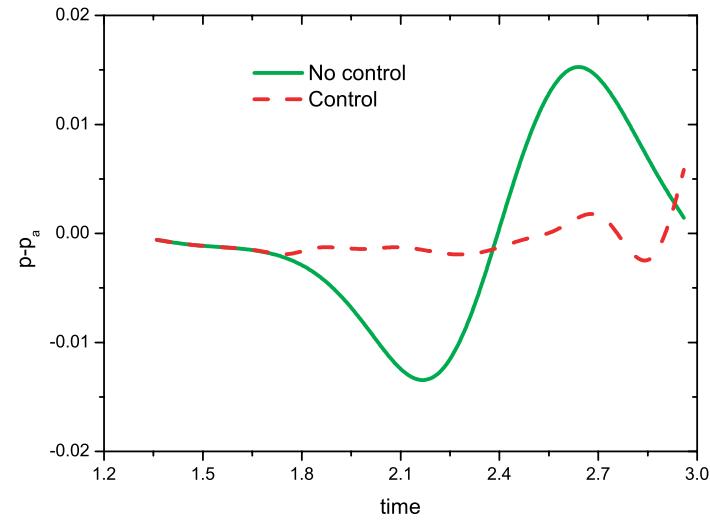
Station 1



Station 2

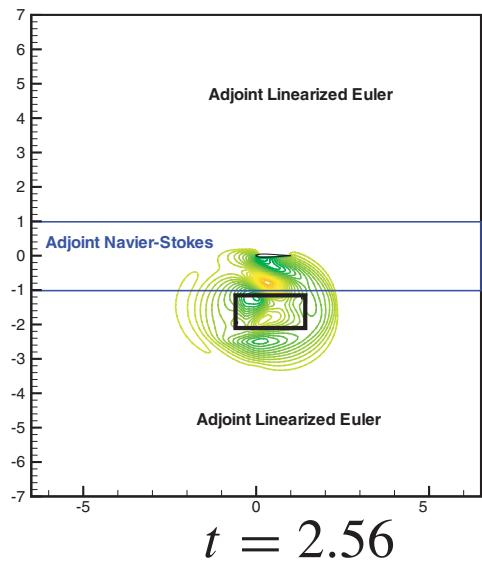


Station 3

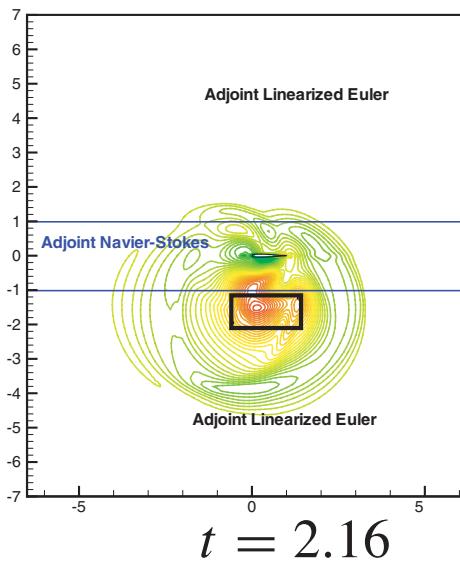


Station 4

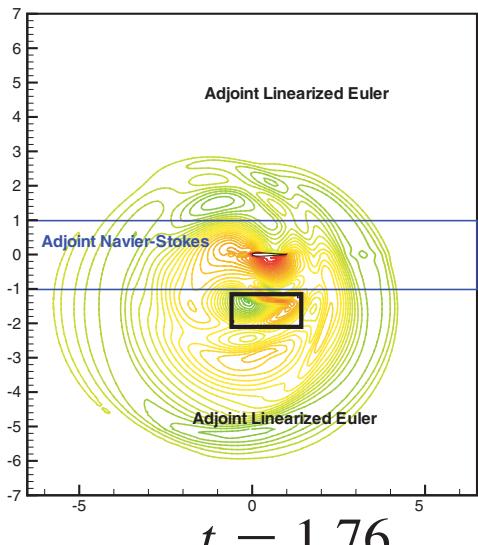
Adjoint Variable λ_4)



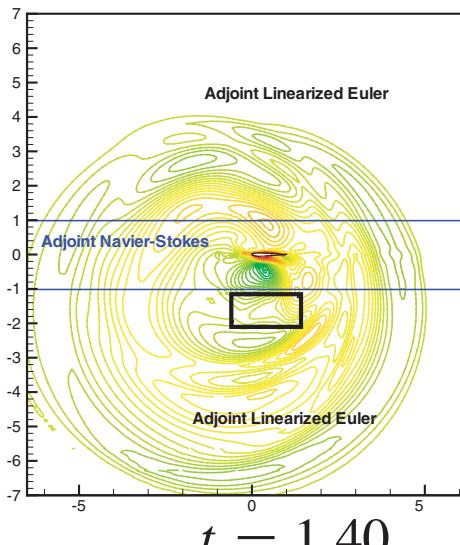
$t = 2.56$



$t = 2.16$



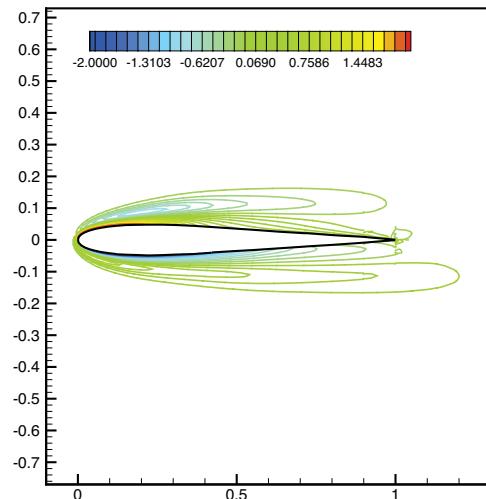
$t = 1.76$



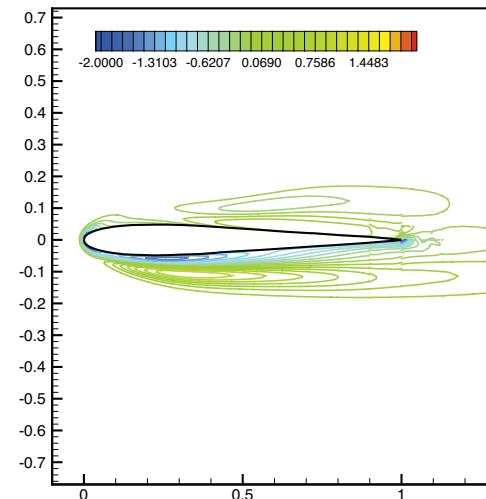
$t = 1.40$



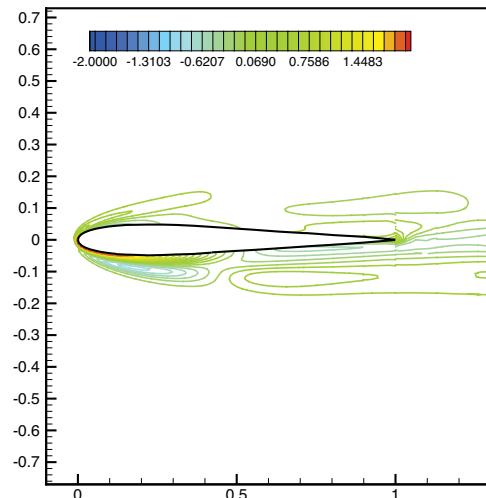
Change in Vorticity



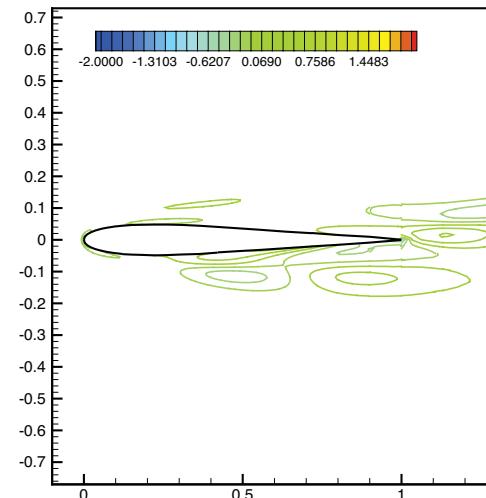
$t = 1.76$



$t = 2.16$

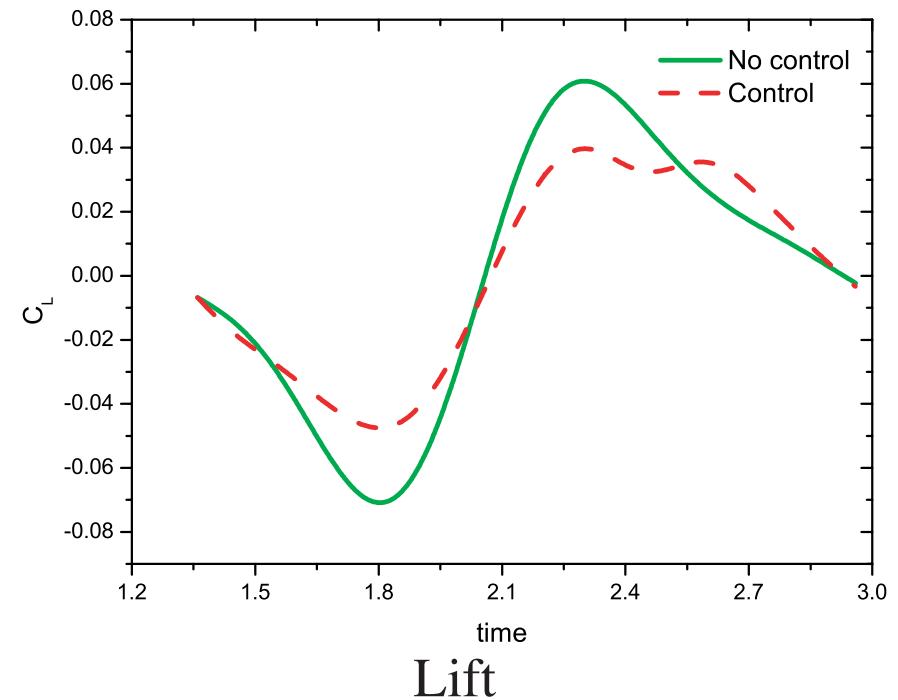
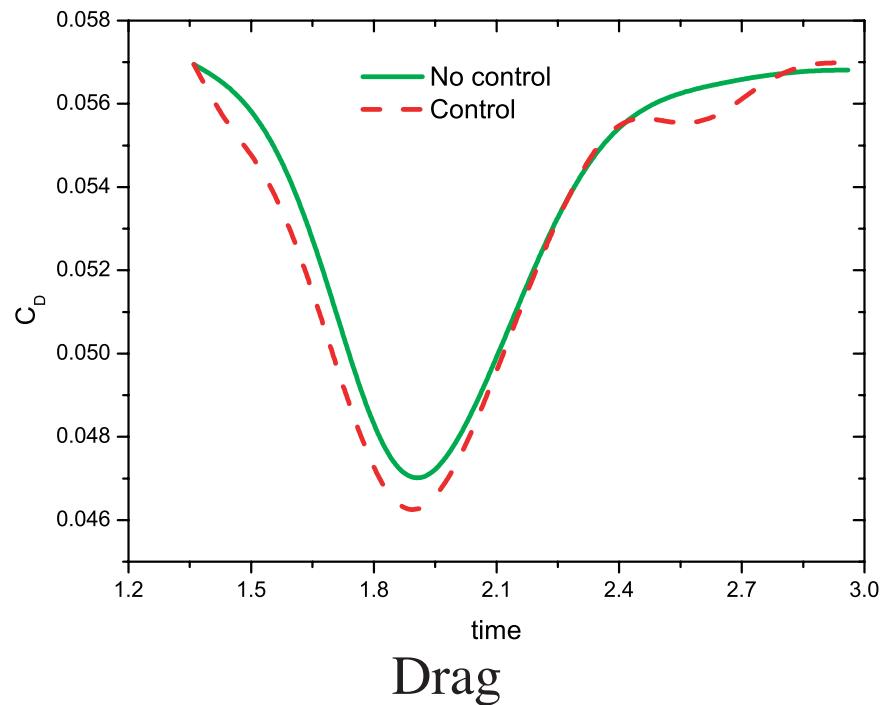


$t = 2.56$



$t = 2.96$

Lift and Drag Time Histories



- Drag is slightly increased in the first half of the optimization time window.
- Peak lift is noticeably reduced.
- Lift gradient is reduced → reduced noise generation.



Closing Comments

- **SAGE** framework is operational for transient, multimodel, multiscale problems with optimization capability.
- Applied **SAGE** to several problems:
 - Acoustic scattering from a circular cylinder
 - Cylinder-vortex interaction
 - Control of vortex shedding
 - Bell AH-1 rotor vortex interaction
- Formulated and implemented multimodel optimization capability (i.e. near-field/far-field coupled adjoint methods)
- Successfully applied multimodel optimal control for BVI model problem:
12db noise reduction.
- **SAGE** is ready for use in simulation and optimization of other (closed loop) flow control systems...
- Sandia provides extensive *enabling technologies* for a wide range of simulation and modeling!

Discontinuous Galerkin Method

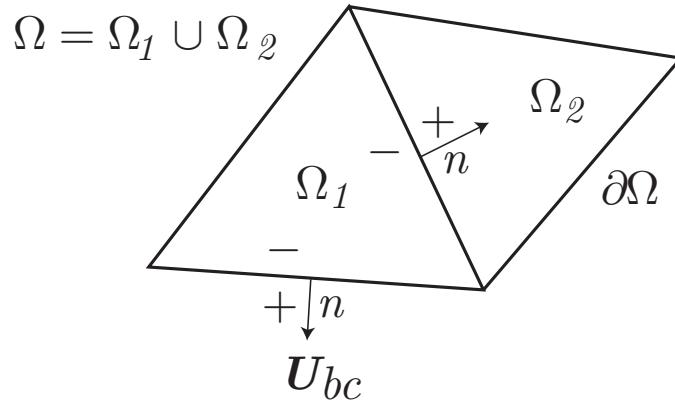
Strong form:

$$\mathbf{U}_{,t} + \mathbf{F}_{i,i} - \mathbf{F}_{i,i}^v = \mathbf{S}, \quad \text{in } \Omega$$

$$\mathbf{U}(x, 0) = \mathbf{U}_0(x), \quad \text{at } t = 0$$

and appropriate boundary conditions on $\partial\Omega$.

Partition Ω into N subdomains Ω_e .



$$\int_{\Omega_e} \left(\mathbf{W}^T \mathbf{U}_{,t} + \mathbf{W}_{,i}^T (\mathbf{F}_i^v - \mathbf{F}_i) \right) dx + \int_{\partial\Omega_e} \mathbf{W}^T (\mathbf{F}_n - \mathbf{F}_n^v) ds = \int_{\Omega_e} \mathbf{W}^T \mathbf{S} ds$$

Introduce numerical fluxes $\mathbf{F}_n(\mathbf{U}) \rightarrow \widehat{\mathbf{F}}_n(\mathbf{U}^-, \mathbf{U}^+)$ and sum over all elements

$$\begin{aligned} & \sum_{e=1}^N \int_{\Omega_e} \left(\mathbf{W}^T \mathbf{U}_{,t} + \mathbf{W}_{,i}^T (\mathbf{F}_i^v - \mathbf{F}_i) \right) dx + \\ & \sum_{e=1}^N \int_{\partial\Omega_e} \mathbf{W}^T (\widehat{\mathbf{F}}_n(\mathbf{U}^-, \mathbf{U}^+) - \widehat{\mathbf{F}}_n^v(\mathbf{U}^-, \mathbf{U}^+)) ds = \sum_{e=1}^N \int_{\Omega_e} \mathbf{W}^T \mathbf{S} ds \quad \forall \mathbf{W} \in \mathcal{V} \end{aligned}$$

Benefits: High accuracy, unstructured, local hp -refinement, local conservation, ...

Note: boundary conditions (and controls) set weakly through numerical fluxes ...



Numerical Inviscid Flux

$$\widehat{\mathbf{F}}_n(\mathbf{U}^-, \mathbf{U}^+) \approx \mathbf{F}_n(\mathbf{U})$$

Monotone property of numerical inviscid flux:

- Consistent with true flux: $\widehat{\mathbf{F}}_n(\mathbf{U}, \mathbf{U}) = \mathbf{F}_n(\mathbf{U})$
- A nondecreasing function of \mathbf{U}^-
- A nonincreasing function of \mathbf{U}^+

Specific Choices:

- Lax-Friedrichs, Steger-Warming, vanLeer, Roe, etc...
(see e.g., Toro 1999 and Cockburn 1999)
- Example: *Lax-Friedrichs flux*

$$\widehat{\mathbf{F}}_n(\mathbf{U}^-, \mathbf{U}^+) = \frac{1}{2} [\mathbf{F}_n(\mathbf{U}^-) + \mathbf{F}_n(\mathbf{U}^+) + \lambda_m (\mathbf{U}^- - \mathbf{U}^+)]$$

where λ_m is the maximum, in absolute value, of the eigenvalues of the Flux Jacobian $\mathbf{A}_n = \partial \mathbf{F}_n / \partial \mathbf{U}$



Numerical Viscous Flux

$$\widehat{\mathbf{F}}_n^v(\mathbf{U}^-, \boldsymbol{\sigma}^-, \mathbf{U}^+, \boldsymbol{\sigma}^+) \approx \mathbf{F}_n^v(\mathbf{U}, \nabla \mathbf{U})$$

- Bassi-Rebay, Baumann-Oden, local-DG, interior-penalty ...
(see e.g., Arnold *et al.* 2001 and Cockburn 1999)
- Example: *Bassi-Rebay flux*
First compute a “jump savvy” gradient of the state, $\boldsymbol{\sigma}_j \sim \mathbf{U}_{,j}$ by solving for

$$\sum_{e=1}^N \int_{\Omega_e} \mathbf{W}^T \boldsymbol{\sigma}_j \, dx = - \sum_{e=1}^N \int_{\Omega_e} \mathbf{W}_{,j}^T \mathbf{U} \, dx + ds + \sum_{e=1}^N \int_{\partial \Omega_e} \mathbf{W}^T \widehat{\mathbf{U}}(\mathbf{U}^-, \mathbf{U}^+) n_j \, ds$$

$\forall \mathbf{W} \in \mathcal{V}$ and $j = 1, 2, 3$, where

$$\widehat{\mathbf{U}}(\mathbf{U}^-, \mathbf{U}^+) = \frac{1}{2} (\mathbf{U}^- + \mathbf{U}^+)$$

The Bassi-Rebay numerical flux is then computed using

$$\widehat{\mathbf{F}}_n^v(\mathbf{U}^-, \boldsymbol{\sigma}^-, \mathbf{U}^+, \boldsymbol{\sigma}^+) = \frac{1}{2} (\mathbf{F}_n^v(\mathbf{U}^-, \boldsymbol{\sigma}^-) + \mathbf{F}_n^v(\mathbf{U}^+, \boldsymbol{\sigma}^+))$$